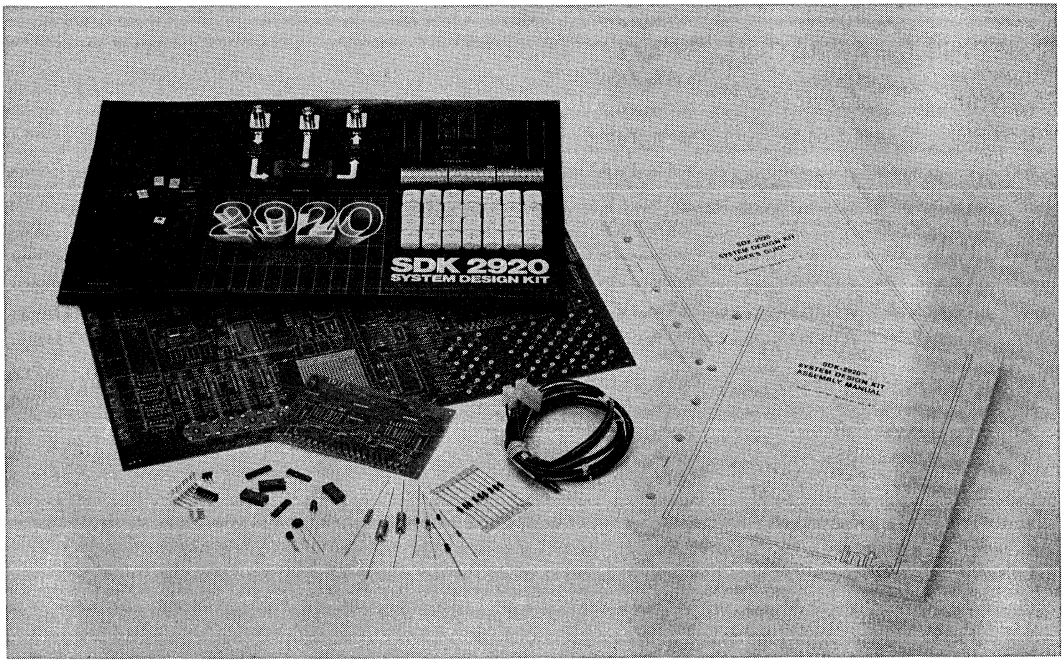
SDK-2920
2920 SYSTEM DESIGN KIT

- **Complete 2920 program development:**
  - **2920 assembly language keyboard**
  - **Single-line assembler/disassembler**
  - **24-character, alphanumeric display**
  - **2920 memory display and modify**
  - **List program memory to line printer with symbol table**
- **Decimal-to-binary conversion program**
- **File handling capabilities:**
  - **Up/down load of object file to Intellec or audio cassette**
  - **Up load source file with symbol table to Intellec**
  - **2920 EPROM programming**
- **Real-time execution of a programmed 2920**
- **Breadboarding area**

The SDK-2920 contains all of the components required to assemble a complete single board microcomputer system for programming and evaluation of the 2920 Analog Signal Processor. The 8085/8041A microcomputer-based program development section allows you to immediately enter programs in 2920 assembly mnemonics, translate them to 2920 object code, and program the on-board 2920 EPROM. The kit supports basic filing options such as up/down loading to/from an Intellec, audio cassette, and line printer. The SDK-2920 also provides the user with a 2920 run mode section allowing real-time execution of a programmed 2920. This section comes complete with BNC connectors and Intel's 2912 PCM line filters required for one input and one output network. The kit supports optional input and output circuitry on the run mode section.



FUNCTIONAL DESCRIPTION

The SDK-2920 is a kit which includes all the necessary hardware and software components to assemble, using common laboratory tools and test equipment, a complete single board 2920 programming and evaluation aid. Assembly generally requires 4 to 8 hours, depending on the experience of the user.

The 2920 Signal Processor

The Intel® 2920 Signal Processor is a programmable, single chip analog and digital signal processor specifically designed to replace analog subsystems in real-time processing applications. Its instruction set plus the high precision (25 bits) digital arithmetic logic unit provides the capability to implement very complex subsystems. Typical functions performed by the 2920 include: lowpass and bandpass filters with up to 20 complex pole and/or zero pairs; threshold detectors; limiters; rectifiers; up to 25-bit multiplication and division; approximations to nonlinear functions such as square law and logarithm; logical operations; input and output multiplexing of signals; logical outputs for decision type processing; and analog outputs for multifrequency oscillators, waveform generators, etc. In addition, several 2920's may be cascaded for very complex processing applications with no loss in throughput rate.

Tables 1 and 2 show the 2920 instruction set and op codes.

Table 1. Shift Op Codes

Operation	Mnemonic	Op Code				Scale Factor
		3	2	1	0	
Shift Right 13 Bits	R13	1	1	0	0	2^{-13}
Shift Right 12 Bits	R12	1	0	1	1	2^{-12}
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
Shift Right 1 Bit	R01	0	0	0	0	2^{-1}
No Shift	R00	1	1	1	1	1
Shift Left 1 Bit	L01	1	1	1	0	2
Shift Left 2 Bits	L02	1	1	0	1	4

System Software

A compact but powerful system monitor is contained in 6K bytes of preprogrammed ROM. The monitor includes system utilities such as command interpretation, user program debugging, and interface controls.

The monitor ROM devices also include a single-line assembler and disassembler. The assembler lets you enter programs in 2920 assembly language mnemonics. The disassembler supports debugging by letting you look at or change either hexadecimal values or 2920 instructions during program interrogation.

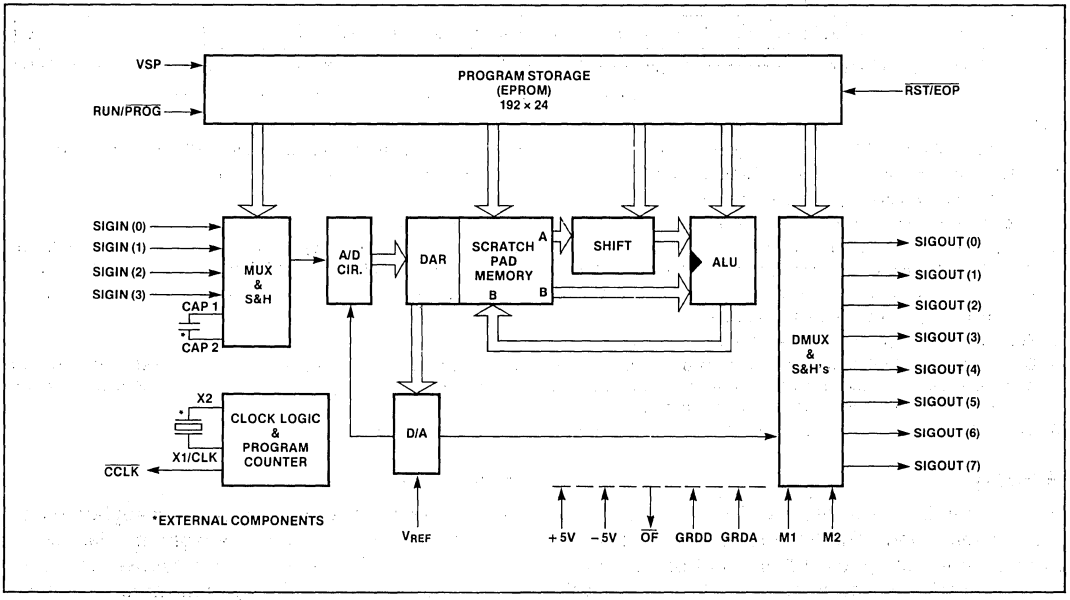
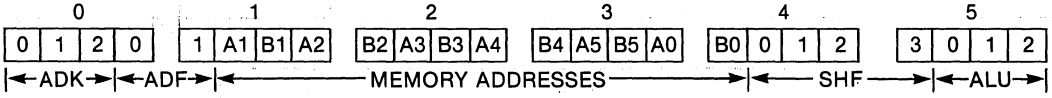


Figure 1. 2920 Function Block Diagram (Run Mode)

Table 2. Instruction Set and Op Codes

Mnemonics	Op Codes[1]			Operations	Notes
Code Condition	**ALU**	**ADF**	**ADK**		
Digital Instructions	**2,1,0**	**1,0**	**2,1,0**		
ADD	110	↑	↑	$(A \times 2^N) + B \rightarrow B$	
SUB	101			$B - (A \times 2^N) \rightarrow B$	
LDA	111			$(A \times 2^N) + 0 \rightarrow B$	
XOR	000			$(A \times 2^N) \oplus B \rightarrow B$	
AND	001			$(A \times 2^N) \cdot B \rightarrow B$	
ABS	011			$\lvert(A \times 2^N)\rvert \rightarrow B$	
ABA[11]	100	[3]	[3]	$\lvert(A \times 2^N)\rvert + B \rightarrow B$	
LIM	010			Sign(A) → ± F.S. → B[4]	
ADD CND( )[2]	110			$(A \times 2^N) + B \rightarrow B$ IFF DAR(K) = 1 $B \rightarrow B$ IFF DAR(K) = 0	
SUB CND( )[2,8]	101			$B - (A \times 2^N) \rightarrow B$ & CY → DAR(K) IFF $CY_P = 1$ $B + (A \times 2^N) \rightarrow B$ & CY → DAR(K) IFF $CY_P = 0$	[5]
LDA CND( )[2]	111			$(A \times 2^N) \rightarrow B$ IFF DAR(K) = 1 $B \rightarrow B$ IFF DAR(K) = 0	
ABA[11] CND( )[9]	100			$(A \times 2^N) + B \rightarrow B$	
XOR CND( )[9]	000			$(A \times 2^N) \oplus B \rightarrow B$	
Analog Instructions					
IN(K)	↑	00	0-3	Signal sample from input channel K	
OUT(K)		10	0-7	D/A to output channel K	
CVTS		00	6	Determine sign bit	
CVT(K)		01	0-7	Perform A/D on bit K	
EOP	[7]	00	5	Program counter to zero	[6]
NOP		00	4	No operation	
CND(K)		11	0-7	Select bit K for conditional instructions	
CNDS	↓	00	7	Select sign bit for conditional instructions	



Note: The input pins for each nibble bit from left to right are D0, D1, D2, D3.

NOTES:

1. Op codes ALU and ADF are in binary notation, ADK is in decimal notation and represents the value "K" when appropriate.
2. CND( ) can be either CND(K) or CNDS testing amplitude bits or the sign bit of the DAR respectively.
3. Determined by analog instructions below.
4. B is set to full scale (F.S.) amplitude with the same sign as the "A" port operand.
5. The previous carry bit (CY_P) is tested to determine the operation. The present carry bit (CY) is loaded into the Kth bit location of the DAR. "Present carry (CY) is generated independent of overflow. It will represent the carry (CY) of a calculated 28-bit result."
6. EOP will also enable overflow correction if it was disabled during a program pass. The EOP must occur in ROM location 188.
7. Determined by digital instructions above.
8. For SUB CNDS Operation $\overline{CY} \rightarrow$ DAR(S).
9. Does not affect DAR. In this case, CND Is used with XOR/ABA to enable/disable the ALU overflow saturation algorithm. Use of either instruction causes the ALU output to roll over rather than go to full scale with sign bit preserved. An EOP instruction will also enable the ALU overflow saturation algorithm.
10. Clarification of CY_{OUT} sense for certain operations. For LDA, XOR, AND, ABS; $CY_{OUT} \rightarrow 0$.

Memory

The kit contains 1.25K bytes of RAM for 2920 program development. The RAM is used as 2920 program memory for up to a 192-instruction 2920 program. The RAM space is also used for a symbol table up to 40 user defined symbols.

User Interface

The kit includes a function and hex keyboard and a formatted 24-character, 18-segment display for easy 2920 code entry. The interactive keyboard and display enables the system monitor to step the user through a command entry sequence with

the friendliness of a menu-driven operating system.

Optional Interfaces

An RS-232 or 20 mA current loop compatible CRT or printer may be used as a listing or file storage device by connecting it to the board's serial interface connector and supplying + 12 and – 12 volts to the board. In addition, the kit provides an audio cassette interface, allowing the use of an audio cassette as a mass storage device.

Debugging

Program development is made easy by use of interactive error messages that will inform the user of illegal entries at the time of program development. Syntax errors are checked for at time of EPROM programming, giving the user the option to continue the programming or not.

The run-mode section allows the user to execute a programmed 2920 in real time, with his own input stimulus and output circuit or instrumentation. The kit is supplied with the 2920-18 and a 5 MHz crystal (800 ns instructions). However, the kit will support the 2920-16 (600 ns instructions) with a 6.67 MHz crystal or clock.

Assembly and Test

The SDK-2920 assembly manual describes assembly in a step-by-step process that includes checking segments of hardware as they are installed. Building the system requires only a few common tools and standard instruments.

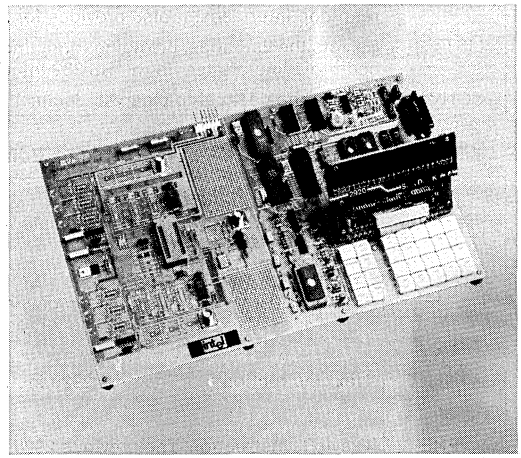


Figure 3. Assembled SDK-2920

CONTROL SECTION

RESET	LIST LOAD
HEX/ASM →	EDIT ←
INSRT NEXT	DEL PREV
SHIFT	CONV CR

2920 DATA KEYS

KP KN	ADD C	ABS D	ABA E	AND F
+/– R	SUB 8	LDA 9	LIM A	XOR B
L .	IN 4	OUT 5	CVTS 6	CVT 7
DAR Y	NOP 0	CNDS 1	CND 2	EOP 3

Figure 2. Keyboard Arrangement

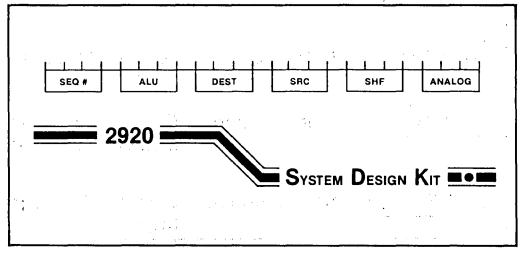


Figure 4. Display Layout

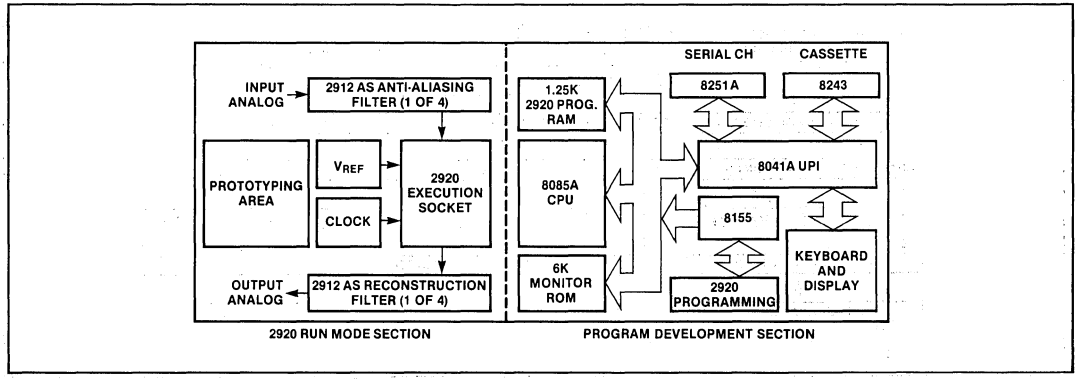


Figure 5. SDK-2920 Functional Block Diagram

Table 3. SDK-2920 Control Commands

RESET	—	Sets the monitor to its initialization program and responds to the selection of one of the four modes. The display will prompt the user with EDIT? LOAD? LIST? CONV?.
SHIFT	—	Selects the upper case characters or functions.
EDIT	—	Selects the edit mode, allowing for 2920 program entry and/or modification. The commands available in the edit mode are shown below in Table 4.
LOAD	—	Selects the load mode, providing for up/down loading to/from the RS-232, cassette, or the 20 mA current loop interfaces. It also provides for 2920 EPROM read, program and verify.
LIST	—	Selects the list mode, providing for listing the 2920 program source code, symbol table, and 2920 hex code to a line printer via the RS-232 interface.
CONV	—	Selects the decimal-to-binary-to-decimal conversion program.

Table 4. Edit Mode Commands

→	Cursor Right	The blinking cursor is moved right one position unless at the end of displayed field.
←	Cursor Left	Blinking cursor is moved left until the sequence number is encountered, then it skips to the left edge of the display.
NEXT	Next Instruction	The next 2920 instruction is displayed unless at end of memory.
PREV	Previous Instruction	The previous 2920 instruction is displayed unless at beginning of memory.
LIST	List Memory	Send disassembled 2920 instructions to serial port.
HEX/ASM	Mode Toggle	Toggle edit mode between symbolic assembly and hexadecimal.
INSRT	Insert Instruction	Expand the program in memory by one location and insert a NOP at current memory display address.
DEL	Delete Instruction	Contract the program in memory by one location and remove the instruction at the current memory display position.

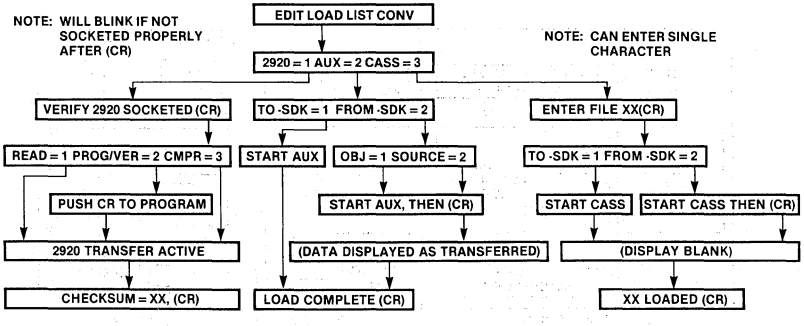


Figure 6. Load Command and Display Tree

SPECIFICATIONS

Control Processor

Intel 8085A microprocessor
6.144 MHz clock rate

Memory

RAM — 1.25K-byte static
ROM — 6K-byte

Interfaces

Keyboard — 28-key with shift, providing 54 functions and characters

Display — 24-character, 18-segment LED

Serial — RS-232 with user-selectable baud rate and 20 mA current loop

Cassette — Hardware and software for audio cassette tape storage interace

Software

System monitor preprogrammed in ROM

2920 assembler and disassembler preprogrammed in ROM

Interface control software preprogrammed in 8041 on-chip ROM

Assembly and Test Equipment Required

- Needle-nose pliers
- Small Phillips screwdriver
- Small flat-blade screwdriver
- Small diagonal wire cutters
- Soldering pencil, <30 watts, 1/16″ diameter tip
- Rosin-core, 60-40 solder, 0.05″ diameter
- Volt-ohm-milliameter, 1 meg ohm input impedance
- Oscilloscope, 1 volt/division vertical sensitivity, 200 μs/division sweep rate, single trace, internal and external triggering

Physical Characteristics

Length — 16 in. (40.64 cm)
Width — 10 in. (25.40 cm)
Height — 4 in. (10.16 cm)
Weight — 3 lb (1.36 kg)

Electrical Characteristics

DC Power Requirements (supplied by user, cables included with the kit)

Program Development Section:

Voltage	Current	Comments
+5V ±5%	1.0A	Required for program development
+12V ±5%	100 mA	Required for 2920 EPROM programming and RS-232 interface
−12V ±5%	100 mA	Required for RS-232 interface

Run Mode Section:

Voltage	Current	Comments
+5V ±5%	300 mA	Required for operation as supplied
	200 mA	Required for each additional 2912/74LS324 pair
−5V ±5%	250 mA	Required for operation as supplied
	200 mA	Required for each additional 2912/74LS324 pair

Environmental Characteristics

Operating Temperature — 0 to 40°C
Relative Humidity — 10% to 90% non-condensing

Reference Manuals

SDK-2920 System Design Kit User's Guide
SDK-2920 System Design Kit Assembly Manual
2920 Analog Signal Processor Design Handbook

ORDERING INFORMATION

Part Number	Description
MCI-20-SDK	2920 System Design Kit